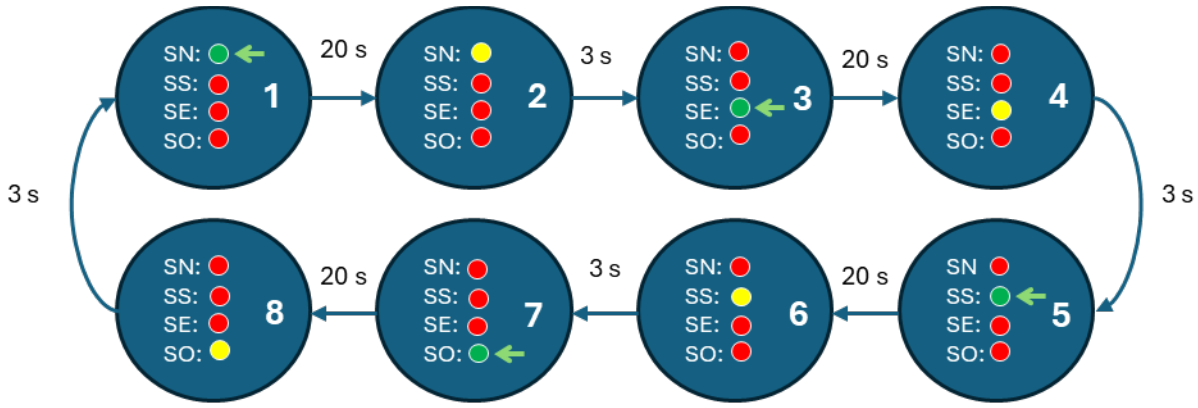
EXAMEN FINAL IMECO FJ24 – 2ª PARTE

Nombre:

INSTRUCCIONES: Para resolver esta parte del examen, utiliza este documento de Word para ir capturando las respuestas correspondientes. Todas tus respuestas deben de estar en este documento, pues es el único que se revisará. Si trabajas en alguna herramienta de apoyo, deberás capturar las imágenes de lo que realizaste y pegarlas en este documento. Sube el archivo como respuesta para que sea revisado posteriormente.

CASO PARA TRABAJAR: Tal como aprendiste en la situación problema para hacer la simulación de un crucero en la ciudad, los autómatas son útiles para representar los estados en que pueden estar los semáforos que conforman el crucero. El siguiente autómata es la modelación de un crucero en el que hay 4 semáforos, y cada uno de ellos tiene 4 luces (rojo, amarillo, verde y flecha izquierda) para controlar el flujo en 4 sentidos, cada uno con posibilidad de vuelta a la izquierda.



SN = Semáforo Norte, SS = Semáforo Sur, SE = Semáforo Este, SO = Semáforo Oeste

Como parte de la implementación del simulador, se ha decidido tener una lista de semáforos del crucero, en el que cada semáforo se representa a su vez con una lista que tiene el identificador del semáforo y la secuencia de luces encendidas según los estados del autómata. Para el autómata que se muestra en la figura anterior, esta sería la lista que lo representa partiendo del estado identificado como el 1, considerando que los símbolos representan a las luces encendidas (r = rojo, a = amarillo, v = verde, f = flecha):

```
(
  (SN (vf a r r r r r r))
  (SS (r r r r vf a r r))
  (SE (r r vf a r r r r))
  (SO (r r r r r r vf a))
)
```

Como puedes observar, el color de la primera luz de las listas internas representa el estado en que se encuentra el autómata. Para el ejemplo anterior, sabríamos que el semáforo Norte (SN) está en verde con su flecha, y el resto de los semáforos están en rojo. El siguiente estado en la simulación está representado en la segunda luz de las listas, indicando que el semáforo Norte cambia a amarillo y los demás permanecen en rojo. Y así sucesivamente.

PROBLEMA I. Implementa en Scheme o Clojure (puedes elegir el lenguaje que prefieras) una función que sirva para generar el siguiente estado de los semáforos de un crucero cuando se da la transición del tiempo. La función recibe como entrada una lista con la representación del autómata como se explicó anteriormente, y genera como resultado una lista con la nueva representación, en la que la primera luz de cada semáforo es la encendida según el siguiente estado del autómata. Considera que el autómata es un ciclo, y por lo tanto, siempre hay un estado que sigue en la simulación.
Por ejemplo, para una entrada a la función con la lista ejemplificada anteriormente, la salida sería la siguiente lista:

```
(
  (SN (a r r r r r r vf))
  (SS (r r r vf a r r r))
  (SE (r vf a r r r r r))
  (SO (r r r r r vf a r))
)
```

Utiliza en tu implementación las herramientas más convenientes (sólo las vistas en clase), para lograr un código elegante y eficiente.

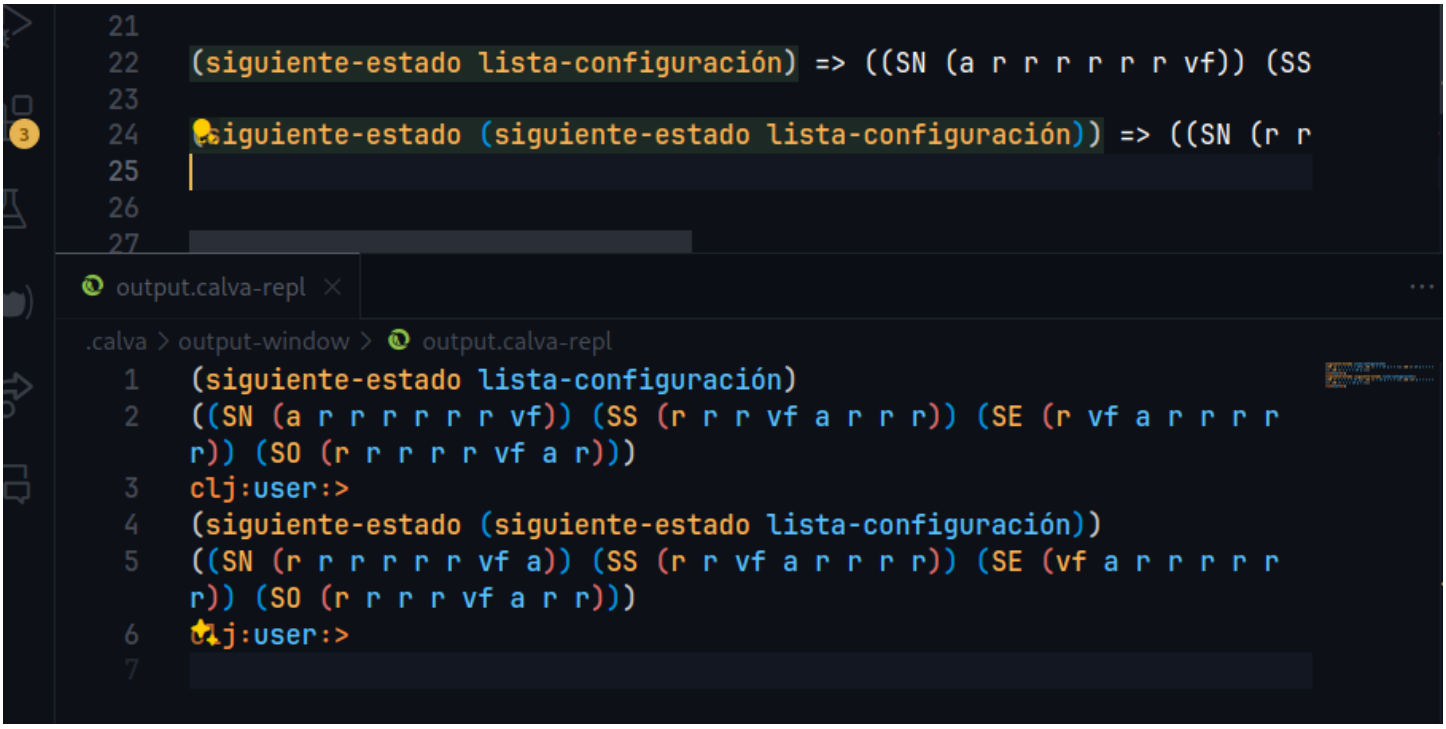
Copia AQUÍ enseguida el código que implementaste y probaste, y una imagen clara de la pantalla con tu código y con las pruebas realizadas para demostrar que funciona.

```
(def lista-configuración '(
(SN (vf a r r r r r r))
(SS (r r r r vf a r r))
(SE (r r vf a r r r r))
(SO (r r r r r r vf a))
))

(def recorre-uno
  (fn [lista]
    (concat (rest lista) (list (first lista)))
    )
  )

(def siguiente-estado
  (fn [lista]
    (map #(list (first %) (recorre-uno (second %))) lista
    )
  )
```

)

(siguiente-estado lista-configuración)

(siguiente-estado (siguiente-estado lista-configuración))



PROBLEMA II. Independientemente de la forma en que implementaste la función anterior, y de si funciona o no, responde la siguiente pregunta en tu documento de Word: ***¿Es posible paralelizar algo en la función que se solicitó en el punto anterior?*** En caso de que tu respuesta sea afirmativa, comenta con qué herramienta de Clojure implementarías la paralelización y justifica brevemente tu respuesta.

Escribe AQUÍ tu respuesta:

Sí, se puede llegar a paralelizar el map que aplico a cada uno de los semaforos , simplemente aplicando un pmap y si se quiere optimizar , un partition para crear un grupo se semaforos, se llegaría a paralelizar esta función

PROBLEMA III. Desde el punto de vista de la teoría de lenguajes que aprendimos en el curso, los semáforos manejan un lenguaje cuyo alfabeto son los símbolos que representan a las luces. En el caso anterior, los símbolos r, a, v y f serían el alfabeto del lenguaje de los semáforos.

a) Utilizando este alfabeto, diseña una **expresión regular** que permita reconocer todas las posibles cadenas de luces en la secuencia de los estados de un semáforo en el autómata que modela un crucero. Notarás que, en la cadena de luces del ciclo, un semáforo puede iniciar con una secuencia de rojos, y sólo en un estado aparece un verde, opcionalmente con flecha, y posteriormente un amarillo, y puede continuar con una secuencia de rojos.
Ejemplos de cadenas válidas para este lenguaje: **rrrvfar**, **rvarrrr**, **var**, **rvfa**, **rrvarrvfa**

Escribe AQUÍ tu respuesta:
((r*v(f|e)ar+)|(r+v(f|e)ar*))+

b) Diseña una **gramática BNF** para reconocer el mismo lenguaje de los semáforos descrito en el inciso anterior.

Escribe AQUÍ tu respuesta:

Secuencia= <rojo><verde><rojo> <secuencia>

Secuencia = e

<rojo>= r <rojo>

<rojo> = e

<verde >= v <vuelta> <amarillo>

<amarillo> = a

<vuelta>= f

<vuelta >=e